**WEEK #4**

**OBJECTIVES**

* To help the students in learning the concepts of C++.
* To help the students in learning about different operators available in C++.
* To help the students in learning the different decision-making statements and control statements used in C++.

**OUTCOMES**

After completing this, the students would be able to:

* Understand the concepts of C++.
* Understand the usage of various operators and their precedence in expression evaluation.
* Understand the usage of decision-making statements and control statements of C++ in real life applications.

**PROBLEMS**

1# Write a C++ program to check whether a number is even or odd using ternary operator.

#include<iostream>

using namespace std;

int main(){

int num;

cout<<"Enter the number: ";

cin>>num;

num%2==0?cout<<"Even":cout<<"Odd";

return 0;

}

2# Write a C++ program to perform the addition of two numbers without using + operator.

#include<iostream>

using namespace std;

int main(){

int num1, num2;

cout<<"Enter the numbes: ";

cin>>num1>>num2;

for(int i=1; i<=num2; i++){

num1++;

}

cout<<"The sum is : "<<num1;

}

3# Write a C++ program to evaluate the arithmetic expression ((a + b / c \* d - e) \* (f - g)). Read the values a, b, c, d, e, f, g from the standard input device.

#include<iostream>

using namespace std;

int main(){

float a, b, c, d, e, f, g;

cout<<"Enter the numbers 'a' to 'g' : ";

cin>>a>>b>>c>>d>>e>>f>>g;

float res = ((a + b / c \* d - e)\* (f - g));

cout<<"The result of the expression is: "<<res;

return 0;

}

4# A Fibonacci sequence is defined as follows: The first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C++ program to generate the first n terms of the sequence.

#include<iostream>

using namespace std;

int main(){

int num;

cout<<"Enter the the number : ";

cin>>num;

int n1 = 0;

int n2 = 1;

cout<<n1<<" ";

cout<<n2<<" ";

for(int i=2; i<=num; i++){

int tmp = n2;

n2= n1+n2;

n1=tmp;

//n2=res;

cout<<n2<<" ";

}

return 0;

}

5# Write a C++ program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.

// C++ program to display Prime numbers till N

#include <bits/stdc++.h>

using namespace std;

// Function to check if a given number is prime

bool isPrime(int n)

{

// Since 0 and 1 is not

// prime return false.

if(n == 1 || n == 0) return false;

// Run a loop from 2 to n-1

for(int i = 2; i < n; i++)

{

// if the number is divisible by i,

// then n is not a prime number.

if(n % i == 0) return false;

}

// Otherwise n is a prime number.

return true;

}

// Driver code

int main()

{

int N;

cout<<"Enter the value of N: ";

cin>>N;

// Check for every number from 1 to N

for(int i = 1; i <= N; i++)

{

// Check if current number is prime

if(isPrime(i))

{

cout << i << " ";

}

}

return 0;

}

6# A character is entered through keyboard. Write a C++ program to determine whether the character entered is a capital letter, a small case letter, a digit or a special symbol using if-else and switch case. The following table shows the range of ASCII values for various characters. Characters ASCII values A – Z: 65 – 90, a – z: 97 – 122, 0 – 9: 48 – 57 Special symbols 0 – 47, 58 – 64, 91 – 96, 123 – 127.

#include<iostream>

using namespace std;

// Printing ASCII character correspoding to a digit

void ascii(int dig){

if(dig>=65 && dig<=90){

cout<<(char)dig;

cout<<" : Entered digit corresponds to upper case character";

}

else if(dig>=97 && dig<=122){

cout<<(char)dig;

cout<<" : Entered digit corresponds to small case character";

}

else if(dig>=48 && dig<=57){

cout<<(char)dig;

cout<<" : Entered digit corresponds to decimal number";

}

else if(dig>=0 && dig<=47 || dig>=58 && dig<= 64 || dig>=123 && dig<=127 ){

cout<<(char)dig;

cout<<" : Entered digit corresponds to special character";

}else{

cout<<"Please enter correct digit.";

}

}

int main(){

int dig;

cout<<"Enter the ASCII digit :";

cin>>dig;

char ch;

ascii(dig);

}

7# Write a C++ program to find the roots of a quadratic equation. 8# Write a C++ program to check whether a given 3-digit number is Armstrong number or not.

#include<iostream>

using namespace std;

int main(){

int num;

cout<<"Enter the number: ";

cin>>num;

int sum = 0;

int tmp=num;

while(tmp!=0){

int rem = tmp%10;

tmp = tmp/10;

sum = sum+rem\*rem\*rem;

}

if(sum==num){

cout<<"The entered number is Armstrong number.";

}else{

cout<<"The entered number is not Armstrong number.";

}

return 0;

}